10/524057
BT01 Rec'd PCT/PTO 08 FEB 2005

In the claims:

1. (original) A method of authenticating a user according to a biometrics parameter of the user presented at an authentication device on a user-presented device on which is stored a biometrics identification template divided into a secure portion and an open portion, the method comprising:

transmitting to a client terminal data derived from said user biometrics parameter at the authentication device;

transmitting from a user-presented device to the client terminal only the open portion of the said biometrics identification template held on the user-presented device;

at the client terminal, implementing a first stage of an identity authentication process between said data and said portion and transmitting the results of said authentication process to the user-presented device; and

at the user-presented device implementing a second stage to complete the identity authentication process using said results and issuing an authentication result based thereon.

2. (original) A method of registration of a user according to a biometrics parameter of the user presented at an authentication device, the method comprising:

transmitting to an authorized client terminal data derived from said user biometrics parameter obtained at the authentication device;

at the authorized client terminal, dividing the biometrics identification template computed into secure portion and open portion,

transmitting from the authorized client terminal to a user-presented device both the open portion and the secure portion of a biometrics identification template,

storing the said template consisting of open and secure portions on the user-presented device, with the secure portion only accessible within the user-presented device and not externally.

3. (originally amended) A method according to claim 1 ~~or 3~~, wherein the secure portion of the biometrics identification template is the portion containing data unauthorized modification of which may cause an impostor to be incorrectly authenticated as a genuine user.

4. (originally amended) A method according to claim 1 ~~or 2~~, wherein the open portion of the biometrics identification template is the portion containing data unauthorized modification of which may not cause an impostor to be incorrectly authenticated as a genuine user.

5. (originally amended) A method according to claim 1~~, 2 or 3~~, wherein the biometrics parameter is a fingerprint.

6. (originally amended) A method according to ~~any preceding~~ claim 1, wherein said open portion of the template comprises parameters of a predetermined number of unique features of the template.

7. (original) A method according to claim 6, wherein the first stage of said identity authentication process

implemented at the client terminal comprises locating unique features using the data derived from the user biometrics parameter and aligning them with said predetermined number of unique features from the identification template held on the user-presented device.

8. (originally amended) A method according to ~~any preceding~~ claim 1, wherein the second stage of the said identity authentication process implemented on the user-presented device is implemented using a local executable matching program stored on the device.

9. (originally amended) A method according to ~~any preceding~~ claim 1, wherein the first stage of the identity authentication process implemented at the client terminal is implemented using a client executable matching program.

10. (original) A method according to claim 9, wherein the client executable matching program is stored on the user-presented device or the authentication device and is transmitted to the client terminal at the time of authentication.

11. (original) A method according to claim 9, wherein the client executable matching program is downloaded by the client terminal from a remote memory at the time of authentication.

12. (originally amended) A method according to ~~any preceding~~ claim 1 wherein the authentication result is used to authenticate a user for authorizing a secure transaction.

13. (original) A method according to claim 12, wherein the secure transaction is controlled by an executable transaction program stored on the user-presented device.

14. (originally amended) A method according to ~~any preceding~~ claim 1, wherein, when the authentication result indicates an adequate match, a first security access check key is constructed including the authentication result.

15. (originally amended) A method according to claim~~s~~ 13 ~~and 14~~, wherein a second security access check key is requested and compared with the first security access key, the result of said comparison being used to enable the executable transaction program if it yields a positive authentication result.

16. (original) A method according to claim 15, wherein the second security access check key is issued from a security server.

17. (originally amended) A method according to ~~any of~~ claim~~s 14 to~~ 16, wherein the first and second security access check keys each include a unique identification number.

18. (originally amended) A method according to claim~~s~~ 15 ~~and 17~~, wherein the unique identification number contains a number obtained from a mathematical operation on a randomly generated number and the authentication result.

19. (original) A method according to claim 18, wherein the randomly generated number changes at each time the number is used.

20. (original) A method according to claim 19, wherein the changing random number is tracked by dividing the number into two portions, a first portion to be used as the current random number and a second portion to be used as the next random number.

21. (originally amended) A method according to claim~~s~~ 17 ~~to 20~~, wherein the unique identification number contains a number that is remembered by the user.

22. (originally amended) A method according to claim~~s~~ 18~~—to 21~~, wherein more than one authentication methods can be used to obtain the authentication result, each being incorporated into the unique identification number.

23. (originally amended) A method according to claim~~s~~ 17~~—to 22~~, wherein the access is divided into several levels and wherein the level of access granted to a user is dependent on the confidence level of positive identity obtained from the unique identification number.

24. (original) A system for authenticating a user according to a biometrics parameter of the user, the system comprising:

a user-presented device on which is stored a biometrics identification template divided into a secure portion and an open portion, wherein only said open portion can be transmitted out of the said device;

an authentication device operable to read biometrics data derived from a user, and comprising means for communicating with the user-presented device and a client terminal;

a client terminal arranged to receive the said open portion of the biometrics identification template held on the user-presented device and the biometrics data derived from the user, and comprising a client processor operable to implement a first stage of and identity authentication process between said data and said portion and to transmit the results of said identity authentication process to the

user-presented device, and wherein the user-presented device comprises a device processor operable to implement a second stage to complete the identity authentication process using said results and to issue an authentication result based thereon.

25. (original) A system according to claim 24, wherein the secure portion of the biometrics identification template is the portion containing data unauthorized modification of which may cause the system to incorrectly authenticate an impostor as a genuine user.

26. (original) A system according to claim 24, wherein the open portion of the biometrics identification template is the portion containing data unauthorized modification of which may not cause the system to incorrectly authenticate an impostor as a genuine user.

27. (original) A system according to claim 24, wherein the biometrics parameter is a fingerprint, and wherein the authentication device includes a fingerprint sensor.

28. (originally amended) A system according to claim 24~~ or 27~~, wherein said portion of the template comprises parameters of a predetermined number of unique features of the template.

29. (originally amended) A system according to claim 24,~~ 27 or 28,~~ wherein the user-presented device comprises a memory in which is stored a local executable matching program for implementing the second stage of the matching process.

30. (original) A system according to claim 29, wherein the memory on the user-presented device stores a client executable matching program which is transmitted to the client processor to implement the first stage of the matching process.

31. (originally amended) A system according to ~~any preceding~~ claim 24, which comprises a security server connected to the client terminal.

32. (original) A system according to claim 31, wherein the security server holds a client executable matching program for implementing the first stage of the matching process.

33. (originally amended) A system according to claim 31 ~~or 32~~, wherein the security server holds a security access check key requestable by the client terminal for enabling a transaction.

34. (originally amended) A system according to ~~any of~~ claim~~s~~ 24 ~~to 33~~, which comprises a transaction server arranged to implement secure transactions and which is in communication with the client terminal so that the authentication result is usable to authenticate a user for authorising a secure transaction.

35. (original) A system according to claim 34, wherein the user-presented device stores an executable transaction program for controlling the secure transaction.

36. (originally amended) A system according to claim~~s~~ 34 ~~and 35~~, wherein more than one authentication methods can be used to obtain the authentication result.

37. (originally amended) A system according to claim~~s~~ 34 ~~to 36~~, wherein the access to the transaction server is divided into several levels and wherein the level of access granted to a user is dependent on the confidence level of positive identity obtained based on the results from the various authentication methods used.

38. (original) A method of executing an operation using first and second processors, the method comprising:

storing in the first processor a first task table containing a plurality of process names with associated process identifiers, each associated with a process locator;

storing in the second processor a second task table containing said of process names and process identifiers;

identifying at the second processor a process to be executed and issuing a request to the first processor to execute said process;

locating said process using the process locator and executing said process at the first processor to generate a result; and

returning the result to the second processor.

39. (original) A method according to claim 38, wherein said process names include object names associated with respective object identifiers.

40. (original) A method according to claim 39, wherein each object has associated therewith a plurality of functions each identified by function names and associated function identifiers in the first and second task tables.

41. (originally amended) A method according to claim 38, ~~39 or 40,~~ wherein the process locator identifies the starting address of a process in a program memory.

42. (originally amended) A method according to ~~any of~~ claim~~s~~ 38~~ to 41~~, wherein the second processor has significantly less processing power than the first processor.

43. (originally amended) A method according to ~~any of~~ claim~~s~~ 38~~ to 42~~, wherein the second processor is arranged to execute locally processes requiring less processing power than those executed by the first processor.

44. (originally amended) A method according to ~~any of~~ claim~~s~~ 38~~ to 43~~, wherein the operation being executed is a fingerprint-matching algorithm comprising a base minutiae finding process executed by the first processor and a minutiae matching process executed by the second processor.

45. (originally amended) A method according to ~~any of~~ claim~~s~~ 38~~ to 44~~, wherein there are a plurality of second processors in communication with a single first processor, each second processor holding a respective task table, and the first processor holding a first task table including all processes identified by the task tables of the second processors.

46. (originally amended) A method according to ~~any of~~ claim~~s~~ 38~~ to 44~~, wherein a client bridge is connected between the first and second processors, the client bridge conveying said requests from the second processor to the first processor and returning the results from the first processor to the second processor.

47. (originally amended) A method according to ~~any of~~ claim~~s~~ 38~~ to 46~~, wherein the first processor is a client terminal and the second processor is embedded on a secure portable computing and data storage platform.

48. (originally amended) A method according to ~~any of~~ claim~~s~~ 38~~ to 44~~, wherein there are a plurality of first processors connected via a client bridge to one or more second processor and arranged to implement different subsets of the processes in the task table of the second processor.

49. (original) A processing system comprising:

a first processor in which is stored a first task table containing a plurality of process names and process identifiers, each associated with a process locator;

a second processor in which is stored a second task table containing said process names with associated process identifiers;

the second processor including a distributed object execution manager for identifying a process to be executed and issuing a request to the first processor to execute said process; and

the first processor including a client distributed object execution manager for controlling the execution of said processes at the first processor, the results of execution of the processes implemented at the first processor being returned to the second processor.

50. (original) A processing system according to claim 49, wherein the first processor includes a client manager for handling communications between the first and second processors.

51. (originally amended) A system according to claim ~~48 or~~ 49, wherein the first processor includes an execution manager for handling the execution of processes.

52. (originally amended) A system according to ~~any of~~ claim~~s~~ 49~~ to 51~~, wherein the first processor comprises a program store for holding said processes, the process locator being used to identify the location of said processes in the program store.

53. (originally amended) A system according to ~~any of~~ claim~~s~~ 49~~ to 52~~, wherein the second processor includes a remote device manager for transmitting said requests to the first processor.

54. (originally amended) A system according to ~~any of~~ claim~~s~~ 49~~ to 53~~, wherein the second processor comprises a stack for holding results returned to it from the first processor.

55. (originally amended) A system according to ~~any of~~ claim~~s~~ 49~~ to 54~~, wherein the second processor includes a program store for holding said processes.

56. (originally amended) A system according to ~~any of~~ claim~~s~~ 49~~ to 55~~, wherein the first processor comprises a client terminal.

57. (originally amended) A system according to ~~any of~~ claim~~s~~ 38~~ to 56~~, which comprises a plurality of first processors, the system further comprising a client bridge for handling communications between the first processors and the second processor.

58. (original) A system according to claim 57, wherein each first processor comprises a server.

59. (originally amended) A system according to claim 57~~ or 58~~, wherein the client bridge includes a network execution

manager for transmitting requests from the second processor to the appropriate one of the first processors, based on a processor identifier in the request.

60. (originally amended) A system according to ~~any of~~ claim~~s~~ 38~~ to 56~~, comprising a plurality of second processors and a client bridge for connecting said second processors to said first processor.

61. (originally amended) A system according to ~~any of~~ claim~~s~~ 38~~ to 60~~, wherein the second or each second processor is embedded on a respective portable secure computing and data storage platform such as smart card.